Secure quantum private information retrieval using phase-encoded queries

Lukasz Olejnik

CERN, 1211 Geneva 23, Switzerland and Poznan Supercomputing and Networking Center, Noskowskiego 12/14, PL-61-704 Poznan, Poland

(Received 14 October 2010; published 8 August 2011)

We propose a quantum solution to the classical private information retrieval (PIR) problem, which allows one to query a database in a private manner. The protocol offers privacy thresholds and allows the user to obtain information from a database in a way that offers the potential adversary, in this model the database owner, no possibility of deterministically establishing the query contents. This protocol may also be viewed as a solution to the symmetrically private information retrieval problem in that it can offer database security (inability for a querying user to steal its contents). Compared to classical solutions, the protocol offers substantial improvement in terms of communication complexity. In comparison with the recent quantum private queries [Phys. Rev. Lett. **100**, 230502 (2008)] protocol, it is more efficient in terms of communication complexity and the number of rounds, while offering a clear privacy parameter. We discuss the security of the protocol and analyze its strengths and conclude that using this technique makes it challenging to obtain the unconditional (in the information-theoretic sense) privacy degree; nevertheless, in addition to being simple, the protocol still offers a privacy level. The oracle used in the protocol is inspired both by the classical computational PIR solutions as well as the Deutsch-Jozsa oracle.

DOI: 10.1103/PhysRevA.84.022313 PACS number(s): 03.67.Dd, 03.67.Ac, 03.67.Lx

I. INTRODUCTION

Database security is a well-studied subject and one that has received huge research attention since the beginning of the information era. This is not unusual, since databases are at the center of many information services and a high level of security is of utmost importance. On the other hand, user-privacy-related problems have not received much attention during the creation of the crucial protocols and information technologies, and that discrepancy between server and user security still exists today. Privacy issues, however, are being raised and these problems are under study, as the importance of the issue has been understood by the community. This problem is even more important because today almost everything is computer driven and the problem of protecting the users' preferences and their habits is of great significance.

Private information retrieval (PIR) [1] is an attempt to formalize the problem of issuing private user queries to a database. It is a two-party protocol where the user side (Alice) wants to obtain some information which is stored in a remote database in another user's (Bob) possession. To make the whole problem more simple, Alice wants to obtain one bit secretly and she knows in advance the address of that bit in Bob's database.

The ideal solution is to simply send the whole database contents to the user and, in fact, this naive attempt is the only one providing information-theoretically with unconditional privacy (with no intractability assumptions). This naive approach is unsatisfactory in terms of the amount of information needed to exchange, and more efficient solutions exist—they, however, assume an increasing number of databases (which also need to be independent). This makes the whole protocol more complicated. Another approach depends on intractability assumptions (computational PIR or CPIR) [2] and further decreases communication complexity as well as enables single-server schemes, which is a major step forward. However, these techniques are inefficient in terms of the number of complex computations and are based on unproven assumptions. One viable extension to this scheme was also discussed in the past where the server's security is also considered: symmetrically private information retrieval (SPIR) [3], where Bob's database is secured against Alice in the sense that she cannot steal it as a whole. This is very important, especially when the contents of the database are valuable. This, however, comes with a price: it further increases the communication complexity of the protocol. The SPIR problem may be perceived as an optimized version of oblivious transfer [4], while in fact the PIR problem implies its existence [5] and the existence of one-way functions [6,7].

The PIR problem also has solutions in the quantum realm, like Refs. [8,9] and, recently, a quantum symmetrically private information retrieval (QSPIR) scheme was proposed; namely, the quantum private queries (QPQ) [10–12], where Alice sends two messages to Bob and has the ability to check his honesty with a nonzero probability, rather than offer privacy. Therefore, it differs significantly from the original PIR.

In this paper we address these problems. We propose a QPIR protocol which gives Alice a chance of user security (i.e., Bob cannot determine Alice's query in a deterministic fashion). The communication complexity is further reduced, as is the whole protocol complexity—it now comprises of a single round. However, it does not offer unconditional security, as this task is still impossible from an information-theoretic point of view. The described protocol may be perceived as a type of quantum private query [10] primitive [13]. Under the same assumptions as with QPQ [14], it may also be considered as an SPIR solution. Considering the single-server solution from [9], our solution differs by the protocol flow: there is only one quantum register in use (only the query part, without the need to store any qubits locally), and no randomness is needed. Thus, our assumptions differ.

The main difference between the QPQ protocol and our approach is that, in QPQ, data are encoded in qubits directly. In contrast, in our approach, the private information is encoded

Index (i)	0	1	2	3
Value ($f_{db}(i)$)	0	1	0	1

FIG. 1. Example database bit string. The values of a database are the results of evaluating a specific boolean function f_{db}.

in the phase. This and the fact that the query is secret for Bob (before he receives it), gives him no deterministic strategy for obtaining Alice's query in a deterministic manner, enhancing the privacy of the protocol. The described protocol may use a boolean transformation, and encodes the private bit in the phase. The protocol has slightly better communication complexity than presented in Refs. [9,10].

II. THE PROTOCOL

A. Data model

The description of the protocol follows: In the PIR problem, Alice wants to perform a query on a database which is in Bob's possession. This database size is $N = 2^n$. Historically, the database in the PIR scheme was modeled as an indexed bit string. Here, this model is inherited with the following modifications: Indices are present—an index is a an address which unambiguously points to a datum. Values, however, are computed by a boolean function $f_{\text{db}} : \{0,1\}^n \to \{0,1\}$. So, assuming a given index i which is about to be queried, the value under that index is $x_i = f_{\text{db}}(i)$, as depicted on Fig. 1.

B. Oracle

In this protocol, Alice sends a single query (encoded in the $|\Psi_1\rangle$ state) to Bob in order to obtain a specified bit. When she does this, Bob is expected to perform unitary operations on the received state and then send that state back to Alice. The oracle is and works as follows:

$$O = \begin{pmatrix} (-1)^{f_{\text{db}}(0)} & & \\ & \ddots & \\ & & (-1)^{f_{\text{db}}(n)} \end{pmatrix}, \tag{1}$$

$$|\Psi_1\rangle = \frac{|0\rangle + |i\rangle}{\sqrt{2}} \xrightarrow{O} |\Psi_2\rangle\,, \tag{2}$$

$$|\Psi_2\rangle = \frac{|0\rangle + (-1)^{f(i)}\,|i\rangle}{\sqrt{2}}. \tag{3}$$

In Eq. (2), the $\frac{|0\rangle+|i\rangle}{\sqrt{2}}$ question state is the superposition of the possible values: the question with a decoy state. This state is submitted by Bob to his oracle [which here is represented by Eq. (2)], implementing the boolean function f_{db} and transformed into a state $|\Psi_2\rangle$, which does contain the actual answer of the query(this is encoded in the phase). It is worth noting the similarity between this oracle and the one known from the Deutsch-Jozsa algorithm [15,16].

C. Query model

How can Alice perform her private query? First, Alice and Bob need to share a communication channel between them to exchange quantum information. To retain security, it is very important for such a channel to be authenticated [17,18]. This is the only assumption we need to have for the protocol to work. To perform a query, Alice sets her n-qubit quantum register R according to the index in Bob's database she wants to know. She sends to Bob the query in the form $|\Psi_1\rangle = \frac{|0\rangle+|i\rangle}{\sqrt{2}}$, where i is the index she wants to securely retrieve, and 0 is another ($i \neq 0$) index in the database. She need to know the answer $f_{\text{db}}(0)$ to the aforementioned boolean function f_{db}, which is x_0. This is very important, as this allows Alice to prepare her measurement and also makes the query private (impossible to overcome in a deterministic fashion). This assumption is equivalent of that in the original QPQ [10], where this special index is dubbed the "rhetoric query." Bob acts on the state $|\Psi_1\rangle = \frac{|0\rangle+|i\rangle}{\sqrt{2}}$ with his oracle. After that, the state will change accordingly to the above-mentioned definition and it will become the answer state. The protocol flow is described on Fig. 2. After the receival of Bob's answer, Alice is in the possession of the aforementioned system $|\Psi_2\rangle$. This system is in two possible states:

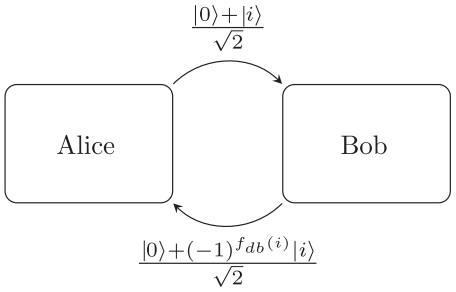

FIG. 2. Diagram describing the behavior of the protocol. Alice is querying for the ith bit (i.e., x_i), which is stored in a database in Bob's possession. To achieve this in a private manner, she encodes her question in a quantum superposition state $|\Psi_1\rangle = \frac{|0\rangle+|i\rangle}{\sqrt{2}}$ and sends this through an authenticated quantum channel. Bob receives the state and uses his oracle, which transforms this state into an answer $|\Psi_2\rangle = \frac{|0\rangle+(-1)^{f_{\text{db}}(i)}|i\rangle}{\sqrt{2}}$ via a conditional phase flip, depending on his function f_{db}, implementing his database. Bob then sends this state to Alice and she has to distinguish between two possible states. It is important for Alice to know the contents of at least one value in Bob's database; the $f_{\text{db}}(0)$, which here has a value of 0. Bob has no possibility of deterministically establishing Alices' query.

$$|\Psi_2\rangle = \begin{cases} \frac{|0\rangle+|i\rangle}{\sqrt{2}}, & f_{\text{db}}(i) = 0 \\ \frac{|0\rangle-|i\rangle}{\sqrt{2}}, & f_{\text{db}}(i) = 1. \end{cases} \tag{4}$$

The logical interpretation for the PIR problem is now straightforward. When the bit's value x_i is 1, the received state is $\frac{|0\rangle-|i\rangle}{\sqrt{2}}$; otherwise, it is $\frac{|0\rangle+|i\rangle}{\sqrt{2}}$. Since those states are obviously orthogonal, it is perfectly possible to distinguish between them by doing a simple projection measurement. Thus, the basic protocol is one round, which differs from the two-round QPQ, and there are fewer registers in use compared to the solutions in Refs. [8,10]. This simplifies the conversation between the parties involved and minimizes potential communication errors.

D. Distinguishing the received answers

The problem of distinguishing these two possible states containing the answer may be solved with the use of m

controlled-NOT (CNOT) gates, assuming that m is the number of ones in Alice's query. Generally, the received state is in an entangled form and, in order to extract classical information from the system, Alice needs to disentangle it. She can perform that by the use of k CNOT gates. Upon receival of the state, Alice performs a unitary transformation

$$\prod_{x=1}^{k} U_{\mathrm{cnot}(1,\mathrm{j_x})} U_{\mathrm{swap}(1,\mathrm{j_x})} \left|\Psi_2\right\rangle = \left|\pm\right\rangle \left|0\right\rangle^{\otimes(n-1)} . \tag{5}$$

The description follows: There are k ones in the state $|\Psi_2\rangle$, with j_x pointing to the first "1," etc. $U_{\mathrm{swap}(1,\mathrm{j_x})}$ is a unitary transformation which swaps between the first and the j_x-nth qubit. This resolves the situation when the first qubit is a zero. The transformation $U_{\mathrm{cnot}(1,\mathrm{j_x})}$ acts as a device that is zeroing every one except that on the first place. Therefore, the received state may be experimentally distinguished by measuring only the first qubit, which will be in the two possible states. The method described here is similar to the one from Ref. [9], but our approach makes use of only a single register (with a swap operation), which makes it simpler.

E. Analysis

We will now discuss and analyze this protocol's properties in detail, particularly against the PIR definition and including Bob's honesty. First, the correctness requirement demands that this protocol be deterministic. This may be considered as a PIR solution only if this necessary condition is met. This protocol obviously satisfies this condition, because the answer is a state which is deterministically distinguishable. The fundamental definition of PIR is thus satisfied.

Second, the security requirement: this protocol gives Alice a 50% chance of making a secure query. This is in contrast with original QPQ [10], where Bob could, in principle, discover the query content with a higher chance and sometimes even evade the (present in QPQ) honesty test—in the generic form of the protocol. These and the possible security improvements are discussed and addressed in Ref. [11].

Third, communication performance is constant, $O(1)$. The exchanged messages are $n + n = 2n$ in size, which is logarithmic of database size.

As for data security, which would be required for the symmetrical security property and—as a result—preventing the data breach, the same argument as with QPQ applies. Bob needs to have some mechanism of controlling (or counting) Alice's queries. Otherwise Alice can simply send all queries one after another and continually steal the whole database. If a resolution to this problem exists and is realizable, this may be considered a quantum symmetrically private information retrieval solution.

The protocol clearly works when Bob is honest. However, let us consider the scenario when Bob is dishonest and tries to prepare his measures in a way to fool Alice. This is a simple intercept and resend attack. It is important to note that he cannot perform the equivalent measurement which Alice does—he does not know which index i she was interested in. Therefore, the only action he may try to undertake is to do a simple projective measurement. He might eventually succeed, but not with more then 50% chance, which is a strict bound for a cheat strategy. However, this risk is still too big compared to the classical setting which is based on the computational intractability assumptions, yet it offers nonzero privacy, while the currently proposed intractability assumptions (in the classical CPIR approach) would not stand against a working quantum computer. On the other hand, here in contrast with the QPQ protocol, Alice does not have any possibility to verify Bob's honesty. But in order to do that, she could periodically check the validity of the received data in, for example, an external database, or just by performing multiple queries to the same database and comparing the outcomes. To some extent this was also the case of QPQ. Bob could not fight against it due to the inability to clone an unknown state [19,20].

The protocol works because of Alice's partial knowledge of the database contents; namely, the value of at least one database element: $f_{\mathrm{db}}(0)$. Bob's inability to craft appropriate measurements prevents him from a deterministic, successful cheating scenario, which is a consequence of the Holevo theorem. It is worth noting that, provided that Alice knows more than this one element $f_{\mathrm{db}}(0)$, she could use these different states in a superposition when performing a repeated query, obtaining multiple consecutive bits in a row. This could potentially increase the security factor against a cheating Bob when using the intercept-resend technique.

F. Extensions

The protocol, in its basic form, allows a private query resulting in a single information bit. As with classical PIR, this can be easily extended for a case with retrieving blocks of data $\{0,1\}^k$. The only apparent solution would be to execute this protocol k times to achieve this; namely, to send a $\prod_{i=0}^{k}(\frac{|0\rangle+|i\rangle}{\sqrt{2}})$ (i.e., multiple different queries), which will result in k secret bits. It is important to note that these queries need to be performed consecutively. In other words, in this mode, the protocol needs to be executed several times. Otherwise, it would be possible for Bob to devise a cheating strategy. However, similar cheating strategies as described above apply (for the single-bit case, Bob may not only try to use the intercept-resend attack, but also other sophisticated scenarios such as entangling the response with qubits in his possession). Thus, this should be perceived as a mode of operation rather than a security improvement.

One other extension would be to add the capabilities to detect a cheating Bob. This could be easily accomplished by employing a technique used in the QPQ [10] protocol, by introducing a decoy state understood as an "honesty certificate," which may be simply achieved by sending two identical states. However, it is important to note that, in this case, Bob would have a greater chance of establishing Alice's query. One could overcome this by introducing an approach based on classical information-theoretic solutions—by introducing two independent servers and comparing the outcomes.

III. CONCLUSION

In this paper we have addressed the well-established problem from theoretical cryptography: private information retrieval. We have described a technique which makes it

possible to perform a secure query to a database using an oracle. Private information is phase-encoded and only a single state is sent as a query. This technique is simpler in comparison with the other recently proposed method: quantum private queries [10] and the one by Kerenidis and de Wolf [8]. Our approach is significantly different and offers privacy and simplicity in addition to the possible extensions which may still offer the ability of checking the honesty. This is a step forward in the state of the art of QPIR.

The problem of an efficient PIR solution is not an easy task to solve, and even taking this to the quantum realm one cannot easily achieve unconditional security performance, especially when database security is also considered. It can, however, offer a single-database solution. Classically, this was only possible in scenarios based on intractability assumptions. So, assuming the existence of a full-scale quantum computer, it still offers a viable PIR solution.

[1] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, J. ACM **45**, 965 (1998); C. Cachin, S. Micali, and M. Stadler, *Advances in Cryptology – EUROCRYPT '99* (1999); C. Gentry and Z. Ramzan, *Proc. 32nd ICALP* (2005), pp. 803–815; S. Yekhanin, *Technical Report ECCC TR06-127* (2006).

[2] E. Kushilevitz and R. Ostrovsky, in *Proceedings of the 38th Annual Symposium on the Foundations of Computer Science* (IEEE Computer Society, Washington DC, 1997), p. 364.

[3] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin, J. Comput. Syst. Sci. **60**, 592 (2000).

[4] S. Wiesner, ACM SIGACT News **15**, 78 (1983); M. O. Rabin, How To Exchange Secrets with Oblivious Transfer,' Technical Report TR-81, Harvard Aiken Computational Laboratory, 1981; A. Jakoby, M. Liskiewicz, and A. Madry, Using Quantum Oblivious Transfer to Cheat Sensitive Quantum bit Commmitment. Complexity of Boolean Functions 2006, e-print arXiv:quant-ph/0605150v1.

[5] G. Di, T. Malkin, and R. Ostrovsky, Lect. Notes Comput. Sci. **1807**, 122 (2000).

[6] A. Beimel, Y. Ishai, E. Kuchilevitz, and T. Malkin, in *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing* (ACM, New York, 1999), pp. 89–98.

[7] For a complete survey of the PIR problem, consult the works [21,22], while the practicality of the current classical schemes is discussed in [23].

[8] I. Kerenidis and R. de Wolf, J. Comput. Syst. Sci. **69**, 395 (2004).

[9] I. Kerenidis and R. de Wolf, Info. Proc. Lett. **90**, 109 (2004).

[10] V. Giovannetti, S. Lloyd, and L. Maccone, Phys. Rev. Lett. **100**, 230502 (2008).

[11] V. Giovannetti, S. Lloyd, and L. Maccone, IEEE Trans. Inf. Theo. **56**, 34465 (2010).

[12] F. De Martini, V. Giovannetti, S. Lloyd, L. Maccone, E. Nagali, L. Sansoni, and Fabio Sciarrino, Phys. Rev. A **80**, 010302 (2009).

[13] C. H. Bennett and G. Brassard, *Proc. IEEE Int. Conf. on Computers, Systems and Signal Processing, Bangalore, India* (IEEE, New York, 1984), pp. 175–179.

[14] From now on we treat QPQ as a protocol and not as a primitive.

[15] D. Deutsch and R. Jozsa, Proc. R. Soc. London A **439**, 553 (1992).

[16] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proc. R. Soc. London A **454**, 339 (1998).

[17] C. Crpeau, L. Salvail, *Advances in Cryptology—Proceedings of Eurocrypt '95, May* 1995 (Springer-Verlag), pp. 133–146.

[18] M. N. Wegman and J. L. Carter, J. Comp. Syst. Sci. **22**, 265 (1981).

[19] W. K. Wootters and W. H. Zurek, Nature (London) **299**, 802 (1982).

[20] G. M. D'Ariano and H. P. Yuen, Phys. Rev. Lett. **76**, 2832 (1996).

[21] R. Ostrovsky, W. E. Skeith III, Lect. Notes Comput. Sci. **4450**, 393 (2007).

[22] A. Beimel, *Private Information Retrieval: a Primer*, [www.cs.bgu.ac.il/beimel/Papers/PIRsurvey.ps] (2008).

[23] R. Sion and B. Carbunar, Proc. of Network and Distributed System Security Symposium, 2007.